ELSEVIER

Operations Research Letters 24 (1999) 47–56

operations research letters

Monte Carlo bounding techniques for determining solution quality in stochastic programs

Wai-Kei Mak[a], David P. Morton[b], R. Kevin Wood[c], *

[a] *Department of Computer Science, The University of Texas at Austin, Austin, TX 78712, USA*
[b] *Graduate Program in Operations Research, The University of Texas at Austin, Austin, TX 78712, USA*
[c] *Operations Research Department, Naval Postgraduate School, Monterey, CA 93943, USA*

Received 1 January 1997; received in revised form 1 October 1998

Abstract

A stochastic program SP with solution value z^* can be approximately solved by sampling n realizations of the program's stochastic parameters, and by solving the resulting "approximating problem" for $(\boldsymbol{x}_n^*, z_n^*)$. We show that, in expectation, z_n^* is a lower bound on z^* and that this bound monotonically improves as n increases. The first result is used to construct confidence intervals on the optimality gap for any candidate solution $\hat{\boldsymbol{x}}$ to SP, e.g., $\hat{\boldsymbol{x}} = \boldsymbol{x}_n^*$. A sampling procedure based on common random numbers ensures nonnegative gap estimates and provides significant variance reduction over naive sampling on four test problems. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Programming; Stochastic; Monte Carlo approximations; Statistics; Sampling; Confidence intervals; Variance reduction

1. Introduction

This paper develops, analyzes and computationally tests a new, probabilistic lower bound for stochastic programs. The bound is used to determine the quality of candidate solutions. Specifically, we construct confidence intervals on the corresponding "optimality gap" which is the difference in objective values between a candidate solution and an optimal solution. The bounding methodology is based on solving an approximating problem generated by a Monte Carlo sampling of the random parameters.

We consider a stochastic optimization problem of the form

$$\text{SP} \quad z^* = \min_{\boldsymbol{x}\in X} Ef(\boldsymbol{x}, \tilde{\boldsymbol{\xi}}) \quad \text{with } \boldsymbol{x}^* \in \operatorname*{argmin}_{\boldsymbol{x}\in X} Ef(\boldsymbol{x}, \tilde{\boldsymbol{\xi}}),$$

where f is a real-valued function, $\boldsymbol{x}$ is a vector of decision variables with deterministic feasible set X, and $\tilde{\boldsymbol{\xi}}$ is a vector of random variables. We are also concerned with an associated *approximating problem*

$$\text{SP}_n \quad z_n^* = \min_{\boldsymbol{x}\in X} \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}, \tilde{\boldsymbol{\xi}}^i) \quad \text{with}$$

$$\boldsymbol{x}_n^* \in \operatorname*{argmin}_{\boldsymbol{x}\in X} \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}, \tilde{\boldsymbol{\xi}}^i),$$

* Corresponding author. Fax: +1 831 656 2595.

where $\tilde{\xi}^i$, $i=1,\ldots,n$, are independent and identically distributed (i.i.d.) from the distribution of $\tilde{\xi}$. Throughout this paper we assume that the first and second moments of $f(\boldsymbol{x},\tilde{\xi})$ exist for all $\boldsymbol{x}\in X$.

Unless the random vector $\tilde{\xi}$ has a small number of possible realizations (also called "scenarios"), it is usually impossible to solve SP exactly. One standard approach for approximately solving SP is to use a Monte Carlo sampling procedure to generate n observations $\tilde{\xi}^i$, $i=1,\ldots,n$, and then solve (a realization of) the approximating problem. This paper is primarily concerned with using approximating problems to determine the quality of a candidate solution $\hat{\boldsymbol{x}}$ to SP. We discuss various methods for generating $\hat{\boldsymbol{x}}$ below.

An important special case of SP, and the one upon which our computational results are based, is the two-stage stochastic LP (linear program) with recourse [3, 36] in which

$$f(\boldsymbol{x},\tilde{\xi}) = \boldsymbol{cx} + \min_{y\geqslant \boldsymbol{0}} \quad \tilde{\boldsymbol{g}}\boldsymbol{y} \qquad \text{s.t.} \quad \tilde{D}\boldsymbol{y} = \tilde{B}\boldsymbol{x} + \tilde{\boldsymbol{d}}, \tag{1}$$

and X is a polyhedral set. Here, $\tilde{\xi} = \mathrm{vec}(\tilde{\boldsymbol{d}},\tilde{\boldsymbol{g}},\tilde{B},\tilde{D})$, where the "vec" operator reads its arguments columnwise to form a single vector. In practice, only a subset of the parameters in $\tilde{\xi}$ are random.

The method of approximately solving SP by solving a realization of SP$_n$ relies on "external sampling", i.e., the sampling is performed external to (prior to) the solution procedure. This approach is justified by the theory of epi-convergence: Under certain assumptions, $\{z_n^*\}_{n=1}^{\infty}$ converges to z^*, w.p.1., and accumulation points of $\{\boldsymbol{x}_n^*\}_{n=1}^{\infty}$ are optimal solutions to SP, w.p.1. See [9, 21, 22, 33] for these results and additional results concerning rates of convergence.

Other Monte Carlo solution procedures for solving SP use "internal sampling". These procedures include sampling-based cutting-plane methods [4, 16, 19] and stochastic quasi-gradient algorithms [10, 11], both of which are adaptations of deterministically valid algorithms in which exact function and (sub-)gradient evaluations are replaced with Monte Carlo estimates. Sampling is "internal" because observations are generated as the algorithm proceeds. Desirable asymptotic properties can be obtained for both internal and external sampling methodologies but, of course, candidate solutions $\hat{\boldsymbol{x}}$ are generated using a finite number of iterations and/or observations. In either methodology, it is difficult to ascertain the quality of such finitely generated candidate solutions.

For the computational results in this paper, we generate a candidate solution $\hat{\boldsymbol{x}}$ by solving approximating problems of the form SP$_n$. However, our methods can also be applied to test the quality of a candidate solution generated by any means such as the algorithms cited above or heuristic procedures.

One obvious approach to testing solution quality is to bound the *optimality gap*, defined as $Ef(\hat{\boldsymbol{x}},\tilde{\xi})-z^*$; this is the approach we take. For two-stage problems, it is straightforward to estimate $Ef(\hat{\boldsymbol{x}},\tilde{\xi})$ using standard statistical procedures. Thus, the optimality gap is easy to bound given a good lower-bound estimator for z^*. Dantzig and Infanger [6] and Higle and Sen [14] use Monte Carlo versions of lower bounds obtained in adaptations of deterministic cutting-plane algorithms. Higle and Sen [17] have also proposed a statistical lower bound that is rooted in duality. These bounds are discussed in greater detail in Section 2.2.

In this paper, we show that $Ez_n^* \leqslant Ez_{n+1}^* \leqslant z^*$. Thus, z_n^* provides a probabilistic lower bound on z^* that improves (in expectation) with increasing sample size. We use this result, in a batch-means procedure, to produce confidence intervals on the optimality gap with respect to any candidate solution $\hat{\boldsymbol{x}}$. The method we propose, like the method of [17], has the advantage that it is independent of specific solution procedures. Our bound estimator is also unbiased, i.e., for any n there is a deterministic constant $\theta_n \leqslant z^*$ with $Ez_n^* = \theta_n$. Consequently, this estimator is well suited for constructing confidence intervals.

Before proceeding, we note that there are two other general approaches that one might consider using for testing or ensuring solution quality for stochastic programs. One approach [34] tests the null hypothesis "the Karush–Kuhn–Tucker (KKT) conditions are satisfied". But, when a stochastic program is solved by a statistical or heuristic procedure, a suboptimal solution is virtually assured and this approach can say little about the quality of such solutions. (Higle and Sen [15], however, derive a bound on the optimality gap that is motivated by the KKT conditions.) Another approach uses the limiting distribution of $\sqrt{n}(\boldsymbol{x}_n^* - \boldsymbol{x}^*)$ to construct confidence intervals for $\boldsymbol{x}^*$. Under certain conditions, asymptotic normality has been verified for the iterates of specific stochastic approximation procedures, for example, the Robbins–Monro procedure;

see the survey in Pflug ([27], Section 5). However, this distribution is usually non-normal because of the constraints $\boldsymbol{x} \in X$ [8, 21, 32]. The method we propose is simple and requires only mild assumptions: We assume that (i) $f(\boldsymbol{x}, \tilde{\xi})$ has finite mean and variance, (ii) i.i.d. observations of $\tilde{\xi}$ can be generated, (iii) instances of SP_n can be solved for sufficiently large n to yield "good" bounding information, and (iv) $f(\boldsymbol{x}, \tilde{\xi})$ can be evaluated exactly for specific values of $\boldsymbol{x}$ and realizations of $\tilde{\xi}$.

In the next section, we review standard Monte Carlo upper (pessimistic) bounds and develop our new Monte Carlo lower (optimistic) bounds. In Section 3, we show how to use these probabilistic bounds to obtain approximate confidence intervals on the quality of a candidate solution; we do this in a naive fashion but also using the variance-reduction technique of common random numbers. Section 4 provides computational results on four two-stage stochastic linear programs from the literature, and Section 5 gives a brief conclusion and mentions areas for further research.

2. Monte Carlo bounds

2.1. Upper bounds

Suppose that we have used a procedure, possibly heuristic, to find a "good", but probably suboptimal solution $\hat{\boldsymbol{x}} \in X$ for a stochastic program SP. We can estimate $Ef(\hat{\boldsymbol{x}}, \tilde{\xi})$ (the expected cost of operating our "system" with a suboptimal decision vector $\boldsymbol{x} = \hat{\boldsymbol{x}}$) via the standard sample mean estimator

$$\bar{U}(n) = \frac{1}{n} \sum_{i=1}^{n} f(\hat{\boldsymbol{x}}, \tilde{\xi}^i),$$

where $\tilde{\xi}^1, \ldots, \tilde{\xi}^n$ are i.i.d. from the distribution of $\tilde{\xi}$. This estimator has two important properties: It is an unbiased estimator of the true cost of a suboptimal decision $\hat{\boldsymbol{x}}$, i.e.,

$$E\bar{U}(n) = Ef(\hat{\boldsymbol{x}}, \tilde{\xi}) \geqslant z^*, \tag{2}$$

and it satisfies the following central limit theorem (CLT):

$$\sqrt{n}[\bar{U}(n) - Ef(\hat{\boldsymbol{x}}, \tilde{\xi})] \Rightarrow \mathrm{N}(0, \sigma_{\mathrm{u}}^2) \text{ as } n \to \infty$$
$$\text{where } \sigma_{\mathrm{u}}^2 = \operatorname{var} f(\hat{\boldsymbol{x}}, \tilde{\xi}). \tag{3}$$

Here, $\Rightarrow$ denotes convergence in distribution, and $\mathrm{N}(0, \sigma^2)$ is a normal random variable with mean zero and variance σ^2. This CLT, coupled with $s_{\mathrm{u}}^2(n)$, the standard sample variance estimator of σ_{u}^2, enables the construction of confidence intervals for $Ef(\hat{\boldsymbol{x}}, \tilde{\xi})$. Unbiasedness and asymptotic normality can also be achieved using variance-reduction techniques such as importance sampling [4, 19].

2.2. Lower bounds

Here we state our main lower-bounding results and discuss related lower bounds from the literature.

Theorem 1. *Let $\tilde{\xi}^1, \ldots, \tilde{\xi}^n$ be i.i.d. from the distribution of $\tilde{\xi}$. Then,*

$$Ez_n^* = E \min_{\boldsymbol{x} \in X} \left[\frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}, \tilde{\xi}^i) \right] \leqslant z^*.$$

Proof.

$$\min_{\boldsymbol{x} \in X} Ef(\boldsymbol{x}, \tilde{\xi}) = \min_{\boldsymbol{x} \in X} E\frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}, \tilde{\xi}^i)$$
$$\geqslant E \min_{\boldsymbol{x} \in X} \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}, \tilde{\xi}^i). \qquad \square$$

Just as Eqs. (2) and (3) lead to confidence intervals on an upper bound on z^*, Theorem 1, exploited in a batch-means approach, will lead to confidence intervals on a lower bound for z^*. Section 3 combines these results to bound the optimality gap. Importantly, Theorem 1 requires little in terms of the structure of SP: $Ef(\boldsymbol{x}, \tilde{\xi})$ must exist $\forall \boldsymbol{x} \in X$, but X need not be convex and $Ef(\boldsymbol{x}, \tilde{\xi})$ need not be convex, unimodal, or smooth. So, the subproblem defined by f could, for example, contain integer decision variables.

We have stated Theorem 1 using i.i.d. samples, but the conclusion of the theorem holds for any unbiased estimator. In particular, $\min_{\boldsymbol{x} \in X} [1/n \sum_{i=1}^{n} f(\boldsymbol{x}, \tilde{\xi}^i)]$ can be replaced with $\min_{\boldsymbol{x} \in X} \mathscr{F}(\boldsymbol{x}, \tilde{\xi}^1, \ldots, \tilde{\xi}^n)$ provided $E\mathscr{F}(\boldsymbol{x}, \tilde{\xi}^1, \ldots, \tilde{\xi}^n) = Ef(\boldsymbol{x}, \tilde{\xi})$. Such generalizations are useful when the lower-bound estimator is constructed using variance-reduction techniques in which the observations are not i.i.d., for example, when using antithetics, stratified sampling, etc., or when $f(\boldsymbol{x}, \tilde{\xi})$ is effectively altered as with importance sampling and some control-variate schemes. (See, for example, Law and Kelton [24] for a discussion of these variance-reduction techniques.) Korf and Wets [23] have similarly extended consistency results for SP$_n$ when the approximating problem is constructed from non-i.i.d. but stationary sequences of observations.

We note that when $n=1$, a special case of the theorem is: $z^* \geqslant E \min_{x\in X} f(\boldsymbol{x}, \tilde{\xi})$. This is the well-known "wait-and-see bound" of Madansky [25]. The intuition behind the wait-and-see bound and the bound of Theorem 1 is similar. In solving the original problem SP, we must find a decision that hedges against all possible realizations of $\tilde{\xi}$. When calculating these lower bounds, we optimize over a subset of $\tilde{\xi}$'s support. Because of this "inside information", we over-optimize and, on average, obtain an optimistic objective value. Based on this same intuition, we expect that the value (and hence the quality) of the bound to grow as n increases. Theorem 2 has been verified independently by Norkin et al. [26].

Theorem 2. *Let $\tilde{\xi}^1, \ldots, \tilde{\xi}^n, \tilde{\xi}^{n+1}$ be i.i.d. from the distribution of $\tilde{\xi}$ and be used to define z_n^* and z_{n+1}^*. Then, $Ez_{n+1}^* \geqslant Ez_n^*$.*

Proof.

$$\begin{aligned} Ez_{n+1}^* &= E \min_{x\in X} \left[\frac{1}{n+1}\sum_{i=1}^{n+1} f(\boldsymbol{x}, \tilde{\xi}^i)\right] \\ &= E \min_{x\in X} \left[\frac{1}{n+1}\sum_{i=1}^{n+1} \frac{1}{n}\sum_{j=1, j\neq i}^{n+1} f(\boldsymbol{x}, \tilde{\xi}^j)\right] \\ &\geqslant \frac{1}{n+1}\sum_{i=1}^{n+1} E \min_{x\in X} \frac{1}{n}\sum_{j=1, j\neq i}^{n+1} f(\boldsymbol{x}, \tilde{\xi}^j) \\ &= Ez_n^*. \quad \square \end{aligned}$$

The monotonicity result of Theorem 2 can also be verified under more general hypotheses. For example, if we use one stream of i.i.d. random vectors, $\tilde{\xi}^1, \tilde{\xi}^2, \ldots$ to define z_n^* and another stream of i.i.d. vectors, say, $\tilde{\xi}'^1, \tilde{\xi}'^2, \ldots$, to define z_{n+1}^*, these streams can contain common random variables (as in the current statement of this theorem), can be independent, or otherwise. Monotonicity is a desirable property because it indicates that, on average, better lower bounds and thus tighter confidence intervals will be obtained as sample sizes increase. The property is not necessary for our computational procedures, however.

There are a number of related results in the literature. Birge [1] introduces a class of deterministic bounds for two-stage stochastic LPs that is based on solving all possible groups of n-scenario problems. In finance, Broadie and Glasserman [2] develop pessimistic and optimistic estimators for the prices of American-style securities; the optimistic bound is found by solving a stochastic dynamic program on a simulated scenario tree. Norkin et al. [26] use pessimistic and optimistic bounds within a branch-and-bound algorithm for stochastic global optimization.

We may reformulate the approximating problem using explicit "non-anticipativity" constraints $\boldsymbol{x}^i = \boldsymbol{x}$ for all i, with associated Lagrange multipliers λ^i, and perform a Lagrangian relaxation of these constraints to obtain

$$\begin{aligned} z_n^* = \min \quad & \left[\frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}^i, \tilde{\xi}^i)\right] \\ \text{s.t.} \quad & \boldsymbol{x}^i \in X, \quad i=1,\ldots,n, \\ & \boldsymbol{x}^i = \boldsymbol{x} : \lambda^i, \quad i=1,\ldots,n, \\ \geqslant \min \quad & \left[\frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}^i, \tilde{\xi}^i) - \lambda^i \boldsymbol{x}^i\right] \\ \text{s.t.} \quad & \boldsymbol{x}^i \in X, \quad i=1,\ldots,n, \end{aligned} \tag{4}$$

provided $\sum_{i=1}^n \lambda^i = \mathbf{0}$. The Monte Carlo lower bound on the right-hand side of Eq. (4) was introduced, via conjugate duality in [17]. The bound is weaker in expectation than Ez_n^*, but the associated optimization problem separates by scenario which is computationally advantageous. Techniques are also proposed in [17] to generate multipliers λ^i and to use the lower-bound estimate to test the quality of a candidate solution $\hat{\boldsymbol{x}}$.

By taking the dual of the second-stage problem, the two-stage stochastic LP (see Eq. (1)) can be rewritten in a manner which suggests the application of Benders decomposition [35]:

$$\begin{aligned} z_n^* = \min_{\boldsymbol{x}\in X} \quad & \boldsymbol{cx} + \theta \\ \text{s.t.} \quad & \theta - \left[\frac{1}{n}\sum_{i=1}^{n} \pi^{k,i}\tilde{B}^i\right]\boldsymbol{x} \geqslant \left[\frac{1}{n}\sum_{i=1}^{n} \pi^{k,i}\tilde{d}^i\right], \\ & k = 1,\ldots,K, \end{aligned} \tag{5}$$

where $(\pi^{k,1},\ldots,\pi^{k,n})$, $k=1,\ldots,K$, denote the extreme points of the Cartesian product of the second-stage dual feasible regions, i.e., $\{\pi : \pi\tilde{D} \leqslant \tilde{\boldsymbol{g}}\}^n$; the corresponding constraints are called "cuts". Cutting-plane algorithms [35] generate a sequence of cuts and lower bounds based on the resulting relaxations. For computational efficiency, Higle and Sen [14] generate a weaker version of these bounds in their stochastic decomposition algorithm by considering a restricted set of dual extreme points. Termination criteria for the stochastic decomposition algorithm are discussed in ([16], Section 5) [14, 15]. Dantzig and Glynn [4] and Dantzig and Infanger [6] consider a related lower bound, and associated termination criteria, using an independent set of observations for each of the cuts instead of the common set of samples that is used in Eq. (5) and in [14].

3. Confidence interval construction and variance reduction

This section develops two approaches for constructing confidence intervals on the optimality gap $Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}}) - z^*$ with respect to a candidate solution $\hat{\boldsymbol{x}}$.

3.1. Independent random number streams

Let $\tilde{\boldsymbol{\xi}}^{i1},\ldots,\tilde{\boldsymbol{\xi}}^{in}$, $i=1,\ldots,n_\ell$, be i.i.d. batches of random vectors. Defining

$$z_n^{*i} = \min_{\boldsymbol{x}\in X} \frac{1}{n}\sum_{j=1}^{n} f(\boldsymbol{x},\tilde{\boldsymbol{\xi}}^{ij}) \quad \text{and} \quad \bar{L}(n_\ell) = \frac{1}{n_\ell}\sum_{i=1}^{n_\ell} z_n^{*i},$$

we have

$$\sqrt{n_\ell}[\bar{L}(n_\ell) - Ez_n^*] \Rightarrow \mathrm{N}(0,\sigma_\ell^2) \quad \text{as } n_\ell \to \infty \\ \text{where } \sigma_\ell^2 = \operatorname{var} z_n^*. \tag{6}$$

Note that the elements $\tilde{\boldsymbol{\xi}}^{ij}$, $j=1,\ldots,n$, within a batch need not be i.i.d. A more general estimator $\mathscr{F}$ can be used to define z_n^{*i} because inter-batch independence is sufficient to ensure that z_n^{*i}, $i=1,\ldots,n_\ell$, are i.i.d. As described after Theorem 1, we simply require an estimator with $E\mathscr{F}(\boldsymbol{x},\tilde{\boldsymbol{\xi}}^1,\ldots,\tilde{\boldsymbol{\xi}}^n) = Ef(\boldsymbol{x},\tilde{\boldsymbol{\xi}})$.

Let $t_{n-1,\alpha}$ satisfy $P\{T_n \leqslant t_{n-1,\alpha}\} = 1-\alpha$, where the random variable T_n has a t distribution with $n-1$ degrees of freedom. Let $s_\ell^2(n_\ell)$ denote the standard sample variance estimator of σ_ℓ^2, let n_u be the number of observations used to estimate $Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}})$, and define

$$\tilde{\varepsilon}_\mathrm{u} = \frac{t_{n_\mathrm{u}-1,\alpha}\, s_\mathrm{u}(n_\mathrm{u})}{\sqrt{n_\mathrm{u}}} \quad \text{and} \quad \tilde{\varepsilon}_\ell = \frac{t_{n_\ell-1,\alpha}\, s_\ell(n_\ell)}{\sqrt{n_\ell}}.$$

In this strategy of "independent random number streams", the upper-bound estimate on z^* is computed using a stream of observations $\tilde{\boldsymbol{\xi}}^1,\ldots,\tilde{\boldsymbol{\xi}}^{n_\mathrm{u}}$ that satisfies Eqs. (2) and (3) and which is independent from the stream used for the lower-bound estimate. Then, for sufficiently large n_u and n_ℓ, we appeal to the respective CLTs (3) and (6). These, coupled with the Boole–Bonferroni inequality (e.g. [24], Section 9.7), Theorem 1, and the fact that $\hat{\boldsymbol{x}}$ is suboptimal, yield

$$\begin{aligned} & P\{\bar{L}(n_\ell) - \tilde{\varepsilon}_\ell \leqslant Ez_n^* \leqslant z^* \leqslant Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}}) \leqslant \bar{U}(n_\mathrm{u}) + \tilde{\varepsilon}_\mathrm{u}\} \\ & \quad \geqslant 1 - P\{\bar{L}(n_\ell) - \tilde{\varepsilon}_\ell \leqslant Ez_n^*\} \\ & \qquad - P\{\bar{U}(n_\mathrm{u}) + \tilde{\varepsilon}_\mathrm{u} \geqslant Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}})\} \\ & \quad \approx 1 - 2\alpha. \end{aligned} \tag{7}$$

From Eq. (7) we may infer that

$$[0, \bar{U}(n_\mathrm{u}) - \bar{L}(n_\ell) + \tilde{\varepsilon}_\ell + \tilde{\varepsilon}_\mathrm{u}]$$

is an approximate $(1-2\alpha)$-level confidence interval for the optimality gap at $\hat{\boldsymbol{x}}$. Due to sampling error we may actually observe $\bar{U}(n_\mathrm{u}) < \bar{L}(n_\ell)$ and hence we recommend the more conservative confidence interval

$$[0, [\bar{U}(n_\mathrm{u}) - \bar{L}(n_\ell)]^+ + \tilde{\varepsilon}_\ell + \tilde{\varepsilon}_\mathrm{u}], \tag{8}$$

where $[y]^+ \equiv \max\{y,0\}$.

Despite the fact that the upper- and lower-bound estimators use independent random number streams, the events $\{\bar{L}(n_\ell) - \tilde{\varepsilon}_\ell \leqslant Ez_n^*\}$ and $\{\bar{U}(n_\mathrm{u}) + \tilde{\varepsilon}_\mathrm{u} \geqslant Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}})\}$ might not be independent, and thus the Boole–Bonferroni inequality is utilized in Eq. (7). Such a situation would arise if $\hat{\boldsymbol{x}}$ is constructed from one or more of the optimal solutions to the n_ℓ lower-bounding problems, for example, via a convex combination of the lower-bounding optimizers. Such averaging procedures are valid provided X is convex and have also been used in stochastic quasi-gradient algorithms to accelerate convergence (e.g. [27], Section 5.1.3).

3.2. Common random number streams

Instead of developing a confidence interval for the optimality gap by estimating $Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}})$ and Ez_n^* separately, as in the "naive sampling strategy" of the previous section, observe that by Theorem 1

$$\underbrace{E\left[\frac{1}{n}\sum_{i=1}^{n} f(\hat{\boldsymbol{x}},\tilde{\xi}^i) - \min_{x\in X}\ \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x},\tilde{\xi}^i)\right]}_{G_n}$$
$$\geqslant Ef(\hat{\boldsymbol{x}},\tilde{\boldsymbol{\xi}}) - z^*. \tag{9}$$

As a result, we may use a batch-means approach to estimate EG_n in which the *same* set of observations is used in the upper- and lower-bound estimators on the left-hand side of Eq. (9). This is an application of the common random numbers (CRN) variance-reduction technique (e.g. [24]).

Note that $G_n \geqslant 0$ so that negative gap estimates cannot arise as they could with the naive sampling strategy; variance reduction over that technique will be obtained if the upper- and lower-bound estimators are positively correlated. The correlation should be high when $\hat{\boldsymbol{x}}$ and $\boldsymbol{x}_n^*$ are "close", and this should occur if $\hat{\boldsymbol{x}}$ is of high quality and the batch size n is sufficiently large. Higle and Sen ([16], Section 5.2.1) have also cited the merits of constructing gap estimators using CRNs.

In analogous fashion to the previous section, let $\tilde{\boldsymbol{\xi}}^{i1},\ldots,\tilde{\boldsymbol{\xi}}^{in}$, $i=1,\ldots,n_{\mathrm{g}}$, be i.i.d. batches. We use each batch to define an observation of the optimality gap, G_n^i, and let $\bar{G}(n_{\mathrm{g}}) = n_{\mathrm{g}}^{-1}\sum_{i=1}^{n_{\mathrm{g}}} G_n^i$. Then,

$$\sqrt{n_{\mathrm{g}}}[\bar{G}(n_{\mathrm{g}}) - EG_n] \Rightarrow \mathrm{N}(0,\sigma_{\mathrm{g}}^2) \quad \text{as } n_{\mathrm{g}} \to \infty$$
$$\text{where } \sigma_{\mathrm{g}}^2 = \operatorname{var} G_n. \tag{10}$$

Let $s_{\mathrm{g}}^2(n_{\mathrm{g}})$ denote the sample variance estimator of σ_{g}^2 and define

$$\tilde{\varepsilon}_{\mathrm{g}} = \frac{t_{n_{\mathrm{g}}-1,\alpha}\, s_{\mathrm{g}}(n_{\mathrm{g}})}{\sqrt{n_{\mathrm{g}}}}.$$

Then, $[0,\bar{G}(n_{\mathrm{g}}) + \tilde{\varepsilon}_{\mathrm{g}}]$ is an approximate $(1-\alpha)$-level confidence interval for the optimality gap at $\hat{\boldsymbol{x}}$.

4. Computational results

This section applies the proposed techniques to solve the four test problems described in Table 1. The first problem, DB1, is a stochastic vehicle-allocation model in a single-commodity network from Donohue and Birge [7]. They use this problem, with fixed first-stage variables, to evaluate a deterministic bound. We allow first-stage variables $\boldsymbol{x}$ to position a fleet of vehicles subject to a "resource constraint" $\boldsymbol{x} \in X = \{\boldsymbol{x} \mid \sum_i x_i = h, x_i \geqslant 0, \forall i\}$. (For ease of solution, we treat first-stage decisions as continuous in all four problems.) The second problem, WRPM1 [5, 19], is an electric power system capacity-expansion model with uncertain demand forecasts and generator reliability. The third problem, 20TERM [13], models a motor freight carrier's operations: First-stage variables position a fleet of vehicles at the beginning of a day while second-stage decisions move the fleet through a multi-commodity network to (i) satisfy point-to-point demands for shipments, and (ii) end the day with a fleet configuration matching the first-stage decision. Penalized violations of second-stage requirements are allowed. The final problem, SSN [31], originates in the telecommunications industry: First-stage variables expand capacity in a communications network, and second-stage variables route demands for point-to-point communication through the network. All four problems have finite discrete distributions for their random parameters. In all cases, the vector $\tilde{\boldsymbol{d}}$ (see Eq. (1)) is stochastic and other problem coefficients are deterministic, except that WRPM1 also includes a stochastic transition matrix $\tilde{B}$ to model unreliable electrical generators.

Table 2 displays computational results for the naive solution strategy based on independent random number streams (see Section 3.1). In calculating the lower bound, $n_\ell = 30$ independent batches are used for each problem. Each batch consists of $n = 25$ i.i.d. observations of the random parameters, except that SSN requires a significantly larger batch size. The upper-bounding estimator is formed using i.i.d. observations that are independent of those used in lower-bound estimation. The upper bound is estimated with respect to a candidate solution $\hat{\boldsymbol{x}} = n_\ell^{-1}\sum_{i=1}^{n_\ell} \boldsymbol{x}_n^{*i}$, where the $\boldsymbol{x}_n^{*i}$ are optimal solutions to the n_ℓ respective approximating problems. The sample size n_{u} used for the upper-bounding estimator is selected to yield an error estimate $\tilde{\varepsilon}_{\mathrm{u}}$ of approximately the same size as $\tilde{\varepsilon}_\ell$ except for WRPM1 where the computational effort would be too great; for this problem we roughly balance the times spent computing lower and upper

Table 1
Test problem descriptions. “rows”, “cols.” and “nonzeros” are, respectively, the numbers of constraints, variables and nonzero constraint entries n the associated problems. $\tilde{B}$ is actually deterministic in three of the problems

Problem	1st stage			2nd stage			Nonzeros in $\tilde{B}\boldsymbol{x}$	Dimension of $\tilde{\xi}$	Total scenarios
	Rows	Cols.	Nonzeros	Rows	Cols.	Nonzeros			
DB1	1	5	5	71	102	255	5	46	4.5×10^{25}
WRPM1	43	75	107	301	289	865	42	11	5.6×10^{6}
20TERM	3	63	63	124	764	4404	63	40	1.1×10^{12}
SSN	1	89	89	175	706	2284	89	86	1.0×10^{70}

Table 2
Test results for sampling with independent random number streams

Problem	DB1	WRPM1	20TERM	SSN
	Lower bound			
Batch size: n	25	25	25	1000
No. of batches: n_ℓ	30	30	30	30
Point estimate: $\bar{L}(n_\ell)$	−17 548	288 865	253 446	9.22
Error estimate ($\alpha = 0.975$): $\tilde{\varepsilon}_\ell$	189	269	885	0.22
CPU minutes (a)	1.1	8.9	34.5	2443
	Upper bound			
Sample size: n_{u}	1000	50 000	20 000	100 000
Point estimate: $\bar{U}(n_{\mathrm{u}})$	−17 661	289 204	254 425	9.98
Error estimate ($\alpha = 0.975$): $\tilde{\varepsilon}_{\mathrm{u}}$	139	722	752	0.11
CPU minutes (b)	0.1	10.1	19.3	120
	Optimality gap			
Point estimate: $[\bar{U}(n_{\mathrm{u}}) - \bar{L}(n_\ell)]^+$	0	338	979	0.76
Confidence interval (95%)	[0,328]	[0,1329]	[0,2616]	[0,1.09]
CPU minutes (a+b)	1.2	19.0	53.8	2563

bounds. As described in Section 3.1, negative gap estimates are possible when the positive-part operator is not applied; note that $\bar{U}(n_{\mathrm{u}}) < \bar{L}(n_\ell)$ for DB1.

Table 3 displays the computational results for sampling based on CRN (see Section 3.2). Unlike the naive strategy, a candidate solution $\hat{\boldsymbol{x}}$ cannot be derived from the approximating problems used to estimate the optimality gap. Therefore, we compute $\hat{\boldsymbol{x}}$ for the CRN strategy by solving an initial approximating problem that has twice as many scenarios as used for lower-bound estimation. The CPU times reported for gap estimation in Table 3 are slightly longer than the lower-bound times in Table 2 because they include (i) the time to solve the 30 approximating problems used for lower-bounding, (ii) the solution time for the initial approximating problem used to generate $\hat{\boldsymbol{x}}$, and (iii) the time for calculating the upper bound terms in G_n (see Eq. (9)). The upper-bound estimates reported in Table 3 are auxiliary calculations that are not necessary to generate the confidence intervals on optimality gaps, but they do allow for a comparison of the two methods of generating candidate solutions $\hat{\boldsymbol{x}}$. The relevant CPU times to compare for the two strategies, for the purpose of determining solution quality, are the “optimality gap” times from the respective tables.

On all problems except SSN, very tight confidence intervals on the optimality gap are obtained with

Table 3
Test results for sampling with common random numbers

Problem	DB1	WRPM1	20TERM	SSN
Sample Size n in SP$_n$ used to generate $\hat{x}$	50	50	50	2000
	Optimality gap			
Batch size	25	25	25	1000
No. of batches: n_g	30	30	30	30
Point estimate: $\bar{G}(n_g)$	23	197	141	0.69
Error estimate ($\alpha = 0.95$): $\tilde{\varepsilon}_g$	5.1	46.3	45.8	0.080
Confidence interval (95%)	[0,28]	[0,243]	[0,187]	[0,0.77]
Variance reduction	4100	460	1300	17
CPU (minutes)	1.2	9.5	37.1	2616
	Upper bound			
Sample size: n_u	1000	50 000	20 000	100 000
Confidence interval (95%)	-17567 ± 114	289186 ± 605	254394 ± 635	10.06 ± 0.12
CPU minutes	0.1	9.9	19.7	119

modest computational effort. The CRN strategy yields confidence interval widths that are 0.2%, 0.08%, 0.07%, and 8% of the upper-bound estimates for DB1, WRPM1, 20TERM, and SSN, respectively. The CRN sampling strategy yields significant computational savings as reflected in the "variance reduction" values calculated as $((\tilde{\varepsilon}_\ell + \tilde{\varepsilon}_u)/\tilde{\varepsilon}_g)^2$ and listed in Table 3. When $\tilde{\varepsilon}_\ell \approx \tilde{\varepsilon}_u$, this quantity gives the approximate multiplicative factor by which sample sizes for the naive strategy must be increased to achieve the confidence interval width of the CRN strategy.

In applying either sampling strategy, we must solve a set of two-stage stochastic LPs. To do so, we use the regularized decomposition (RD) algorithm [29] which is a cutting-plane algorithm whose master program contains a quadratic proximal term. RD tends to converge more quickly than standard cutting-plane methods and can better exploit good starting solutions to speed convergence [30]. In solving DB1, WRPM1, and 20TERM, we apply our implementation of RD, which uses IBM's Optimization Subroutine Library [18] to solve LP subproblems and LSSOL [12] to solve the quadratic master program. Our code cannot solve SSN in a reasonable amount of time, however, so for this problem we use the RD implementation of Ruszczyński and Świetanowski [30]. (As explained below, our code does solve the first three problems faster than the Ruszczyński and Świetanowski code.)

In solving the sequence of approximating problems with our RD algorithm, we accelerate solutions by using the average of the optimal solutions from previous problems as the starting proximal point for the next problem. For comparison, we also solved each of the 30 approximating problems without this enhancement: Running times increase by 25%, 100%, and 330% for DB1, WRPM1, and 20TERM, respectively. It is unfair to compare running times of our code to the times for the Ruszczyński and Świetanowski code because their code uses more stringent stopping tolerances (10^{-8} versus 10^{-4}), and it was not possible to apply the "starting point enhancement" in their code. However, the "lower-bound" times reported in Table 2 for DB1, WRPM1, and 20TERM are faster than their code by factors of 2.2, 9, and 1.5, respectively. Thus, it is likely that the solution time for SSN can be substantially improved.

5. Conclusions and extensions

We have shown that the solution value to a standard approximating problem (SP$_n$) for a two-stage stochastic program (SP) yields a lower bound, in expectation, on the solution value of SP. This result has been exploited, in a batch-means approach, to develop confidence intervals on the optimality gap with respect to any candidate solution to SP. Computational efficiency is improved by using common random number (CRN) streams for gap estimation and by using regularized decomposition.

Because the lower-bounding result is so simple and so general, confidence intervals may be obtained for two-stage stochastic programs with general structure, e.g., with integer first- or second-stage variables, with randomness in any of the second-stage parameters, etc. Although not developed in this paper, the lower-bounding result extends to multi-stage stochastic programs through a straightforward, recursive application of the proof of Theorem 1. The intuition regarding the multi-stage result is similar: If we optimize over a subset of the possible futures at each branch in the scenario tree, we will obtain an optimistic result, on average. Extending the techniques described here to the multi-stage problem is an important area for future research.

The computational efficiency of our proposed methods can certainly be improved. We have examined variance reduction using CRNs, but other techniques should be explored. Regularized decomposition allows us to exploit good starting solutions to solve m instances of SP$_n$ faster than m times the time required to solve a single instance, but other techniques may also speed solutions. For instance, tight cuts from a solution of one instance of SP$_n$ might be used (temporarily) to help solve another instance. Also, m approximating problems can obviously be solved on m parallel processors with a near-linear improvement in speed.

Under certain assumptions, most notably that SP have a unique optimal solution, $\sqrt{n}(z_n^* - z^*)$ is asymptotically normal with mean zero and variance $\operatorname{var} f(\boldsymbol{x}^*, \tilde{\xi})$ [33]. (There are also related results concerning the stochastic order of convergence [28], Section 6). Under this hypothesis, our batch-means approach can be simplified: Confidence intervals on solution quality can be constructed by solving a single approximating problem. Infanger's [20] approach to constructing such confidence intervals in the context of a cutting-plane algorithm shows promise.

Acknowledgements

David Morton's research was partially supported by the Summer Research Assignment program at The University of Texas at Austin and by the National Science Foundation through grant DMI-9702217. Kevin Wood's research was supported by the Air Force Office of Scientific Research and the Office of Naval Research. The authors thank John Birge, Julia Higle, and Suvrajeet Sen for valuable discussions, thank Andrzej Ruszczyński and Artur Świetanowski for access to their regularized decomposition code and thank the associate editor, David Goldsman, for many helpful comments.

References

[1] J.R. Birge, The value of the stochastic solution in stochastic linear programs with fixed recourse, Math. Program. 24 (1982) 314–325.

[2] M. Broadie, P. Glasserman, Pricing American-style options using simulation, J. Econom. Dyn. Control 21 (1997) 1323–1352.

[3] G.B. Dantzig, Linear programming under uncertainty, Manag. Sci. 1 (1955) 197–206.

[4] G.B. Dantzig, P.W. Glynn, Parallel processors for planning under uncertainty, Ann. Oper. Res. 22 (1990) 1–21.

[5] G.B. Dantzig, P.W. Glynn, M. Avriel, J.C. Stone, R. Entriken, M. Nakayama, Decomposition techniques for multi-area generation and transmission planning under uncertainty, EPRI Report 2940-1, Palo Alto, California, 1989.

[6] G.B. Dantzig, G. Infanger, A probabilistic lower bound for two-stage stochastic programs, Technical Report SOL 95-6, Stanford University, Stanford, California, November 1995.

[7] C.J. Donohue, J.R. Birge, An upper bound on the network recourse function, Working Paper, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan, 1995.

[8] J. Dupačová, On non-normal asymptotic behavior of optimal solutions for stochastic programming problems and on related problems of mathematical statistics, Kybernetika 27 (1991) 38–52.

[9] J. Dupačová, R.J.-B. Wets, Asymptotic behavior of statistical estimators and of optimal solutions of stochastic optimization problems, Ann. Statist. 16 (1988) 1517–1549.

[10] Y. Ermoliev, Stochastic quasigradient methods and their applications to systems optimization, Stochastics 9 (1983) 1–36.

[11] A. Gaivoronski, Stochastic quasigradient methods and their implementation, in: Y. Ermoliev, R.J.-B. Wets (Eds.), Numerical Techniques for Stochastic Optimization, Springer, Berlin, 1988.

[12] P.E. Gill, S.J. Hammarling, W. Murray, M.A. Saunders, M.H. Wright, User's guide for LSSOL (version 1.0), Technical Report SOL 86-1, Stanford University, Stanford, California, January 1986.

[13] B. Hackney, G. Infanger, Private communication, June 1994.

[14] J.L. Higle, S. Sen, Stochastic decomposition: an algorithm for two-stage linear programs with recourse, Math. Oper. Res. 16 (1991) 650–669.

[15] J.L. Higle, S. Sen, Statistical verification of optimality conditions for stochastic programs with recourse, Ann. Oper. Res. 30 (1991) 215–240.

[16] J.L. Higle, S. Sen, Stochastic Decomposition: A Statistical Method for Large Scale Stochastic Linear Programming, Kluwer Academic Publishers, Dordrecht, 1996.

[17] J.L. Higle, S. Sen, Duality and statistical tests of optimality for two stage stochastic programs, Math. Programm. 75 (1996) 257–275.

[18] IBM Corporation, Optimization Subroutine Library Guide and Reference, Release 2, Kingston, New York, 1991.

[19] G. Infanger, Monte Carlo (importance) sampling within a Benders decomposition algorithm for stochastic linear programs, Ann. Oper. Res. 39 (1992) 69–95.

[20] G. Infanger, Optimization under uncertainty, OR 372 Class Notes, Stanford University, Stanford, California, Spring 1996.

[21] A.J. King, R.T. Rockafellar, Asymptotic theory for solutions in statistical estimation and stochastic programming, Math. Oper. Res. 18 (1993) 148–162.

[22] A.J. King, R.J.-B. Wets, Epi-consistency of convex stochastic programs, Stochastics 34 (1991) 83–91.

[23] L.A. Korf, R.J.-B. Wets, Ergodic limit laws for stochastic optimization problems, Working Paper, Dept. of Mathematics, University of California, Davis, 1996.

[24] A.M. Law, W.D. Kelton, Simulation Modeling and Analysis, 2nd ed., McGraw-Hill, New York, 1991.

[25] A. Madansky, Inequalities for stochastic linear programming problems, Manag. Sci. 6 (1960) 197–204.

[26] V.I. Norkin, G.Ch. Pflug, A. Ruszczyński, A branch and bound method for stochastic global optimization, IIASA Working Paper 96-065, Laxenburg, Austria, April 1996.

[27] G.C. Pflug, Optimization of Stochastic Models: The Interface Between Simulation and Optimization, Kluwer Academic Publishers, Boston, MA, 1996.

[28] R.Y. Rubinstein, A. Shapiro, Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method, Wiley, Chichester, 1993.

[29] A. Ruszczyński, A regularized decomposition method for minimizing a sum of polyhedral functions, Math. Programm. 35 (1986) 309–333.

[30] A. Ruszczyński, A. Świetanowski, On the regularized decomposition method for two stage stochastic linear problems, IIASA Working Paper 96-014, Laxenburg, Austria, February 1996.

[31] S. Sen, R.D. Doverspike, S. Cosares, Network planning with random demand, Telecommun. Systems 3 (1994) 11–30.

[32] A. Shapiro, Asymptotic properties of statistical estimators in stochastic programming, Ann. Statist. 17 (1989) 841–858.

[33] A. Shapiro, Asymptotic analysis of stochastic programs, Ann. Oper. Res. 30 (1991) 169–186.

[34] A. Shapiro, T. Homem-de-Mello, A simulation-based approach to two-stage stochastic programming with recourse, Math. Programm. 81 (1998) 301–325.

[35] R.M. Van Slyke, R.J.-B. Wets, L-Shaped linear programs with applications to optimal control and stochastic programming, SIAM J. Appl. Math. 17 (1969) 638–663.

[36] R.J.-B. Wets, Stochastic programs with fixed recourse: the equivalent deterministic program, SIAM Rev. 16 (1974) 309–339.